Apply filters to SQL queries Project

MUHAMMAD ALI KHAWAR BUTT

Project description

My organization has been working to make its systems more secure. I use SQL to filter large amounts of data using SQL using filters. These filters include keywords like WHERE and LIKE, etc.

Retrieve after hours failed login attempts

There was a potential security threat that occurred after working hours, after 18:00. I used SQL to filter all the login attempts after 18:00 hours.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_time > '18:00' AND success = FALSE;
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|       18 | pwashing | 2022-05-11 | 19:28:50   | US      | 192.168.66.142  |       0 |
|       20 | tshah    | 2022-05-12 | 18:56:36   | MEXICO  | 192.168.109.50  |       0 |
```

Here, the first portion shows the code and the second portion shows the result.
In the code:
Select *: selects all columns from the table.
FROM log_in_attempts: Specifies the table from which to fetch the data.
Where login_time >: Filters records so only those with the login date having **May 8** or **May 9, 2022,** are shown.

Retrieve login attempts on specific dates

A suspicious activity occurred on the date 2022-05-09. I used SQL to filter the data around these dates.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE login_date = '2022-05-09' OR login_date = '2022-05-08';
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.243.140 |       0 |
|        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.151.162 |       0 |
|        4 | dkot     | 2022-05-08 | 02:00:39   | USA     | 192.168.178.71  |       0 |
```

Here I used the OR operator to filter the data between the dates 2022-05-09 and 2022-05-08.

Retrieve login attempts outside of Mexico

While investigating the organizations data on login attempts , a issue was found with login attempts outside of Mexico.

```
MariaDB [organization]> SELECT *
    -> FROM log_in_attempts
    -> WHERE NOT country LIKE 'MEX%';
+----------+----------+------------+------------+---------+-----------------+---------+
| event_id | username | login_date | login_time | country | ip_address      | success |
+----------+----------+------------+------------+---------+-----------------+---------+
|        1 | jrafael  | 2022-05-09 | 04:56:27   | CAN     | 192.168.243.140 |       0 |
|        2 | apatel   | 2022-05-10 | 20:27:27   | CAN     | 192.168.205.12  |       0 |
|        3 | dkot     | 2022-05-09 | 06:47:41   | USA     | 192.168.151.162 |       0 |
```

We used the NOT operator and the keyword LIKE to include all data entries with the keyword MEX and the % sign to add all characters after it.

Retrieve employees in Marketing

To this end, I need to know which employees' computers need updating. My team has proposed renewing the machines of some employees in the Marketing department.

This is a code snippet that shows how I created a SQL query that filters for employee machines of those employees in the Marketing department situated in the East building.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Marketing' AND office LIKE 'East%';
+-------------+--------------+----------+------------+----------+
| employee_id | device_id    | username | department | office   |
+-------------+--------------+----------+------------+----------+
|        1000 | a320b137c219 | elarson  | Marketing  | East-170 |
|        1052 | a192b174c940 | jdarosa  | Marketing  | East-195 |
|        1075 | x573y883z772 | fbautist | Marketing  | East-267 |
```

Retrieve employees in Finance or Sales

Indeed, the machines for all employees in the Finance and Sales departments need updating. Since these departments require a different security update, I need to fetch information of employees from these two departments only.

The following code demonstrates how I created a SQL query to filter for employee machines from employees in Finance or Sales departments.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE department = 'Finance' OR department = 'Sales';
+-------------+--------------+----------+------------+-------------+
| employee_id | device_id    | username | department | office      |
+-------------+--------------+----------+------------+-------------+
|        1003 | d394e816f943 | sgilmore | Finance    | South-153   |
|        1007 | h174i497j413 | wjaffrey | Finance    | North-406   |
|        1008 | i858j583k571 | abernard | Finance    | South-170   |
```

Retrieve all employees not in IT

One more security update needs to be done for employees not in the Information Technology department, and I need to gather information about those employees prior to making the update.

The example that follows illustrates how I created an SQL query to filter for employee machines from employees outside the Information Technology department.

```
MariaDB [organization]> SELECT *
    -> FROM employees
    -> WHERE NOT department = 'Information Technology';
+-------------+--------------+----------+-----------------+-------------+
| employee_id | device_id    | username | department      | office      |
+-------------+--------------+----------+-----------------+-------------+
|        1000 | a320b137c219 | elarson  | Marketing       | East-170    |
|        1001 | b239c825d303 | bmoreno  | Marketing       | Central-276 |
|        1002 | c116d593e558 | tshah    | Human Resources | North-434   |
```

We used the NOT operator to remove the IT department

Summary

I applied filters to SQL queries to get specific information on login attempts and employee machines. I used different tables and used the AND OR and NOT operators to filter for the specific information needed for each task. I also used LIKE and the percentage sign (%) wildcard to filter for patterns.